Rhythm-based segmentation of Popular Chinese Music

Jensen, Karl Kristoffer

Published in:
Proceeding of the ISMIR

Publication date:
2005

Document Version
Early version, also known as pre-print

RHYTHM-BASED SEGMENTATION OF POPULAR CHINESE MUSIC

Kristoffer Jensen
Department of Medialogy
University of Aalborg Esbjerg
Niels Bohrsvej 8
6700 Esbjerg, Denmark
krist@cs.aaue.dk

Jieping Xu
School of Information
Renmin University
Beijing, China
jieping.xu@263.net

Martin Zachariasen
Department of Computer Science
University of Copenhagen
Universitetsparken 1
2100 Copenhagen, Denmark
martinz@diku.dk

ABSTRACT

We present a new method to segment popular music based on rhythm. By computing a shortest path based on the self-similarity matrix calculated from a model of rhythm, segmenting boundaries are found along the diagonal of the matrix. The cost of a new segment is optimized by matching manual and automatic segment boundaries. We compile a small song database of 21 randomly selected popular Chinese songs which come from Chinese Mainland, Taiwan and Hong Kong. The segmenting results on the small corpus show that 78% manual segmentation points are detected and 74% automatic segmentation points are correct. Automatic segmentation achieved 100% correct detection for 5 songs. The results are very encouraging.

Keywords: Segmentation, Shortest path, Self-similarity, Rhythm

1 INTRODUCTION

Segmentation has a perceptual and subjective nature. Manual segmentation can be due to different dimensions of music, such as rhythm or harmony. Measuring similarity between rhythms is a fundamental problem in computational music theory. In this work, music segmentation is based solely on rhythm, which can be perceived by a changing beat or different instrument modulation. While rhythm is indeed an important music feature, parts of other features, such as loudness and timbre, are also present in the rhythm information, because of the way it is calculated. However, harmony information is not. In Chinese music, rhythm is sometimes different compared to Western music. The beat is not clearly present in some music styles, and the rhythm boundaries depend mainly on the voice.

Segmentation of music has many applications such as music information retrieval, copyright infringement resolution, fast music navigation and repetitive structure finding.

Music segmentation is a popular topic in research today. Several authors have presented segmentation and visualization of music using a self-similarity matrix [1-3] with good results. Foote [2] used a measure of novelty calculated from the self-similarity matrix. Jensen [1] optimized the processing cost by using a smoothed novelty measure, calculated on a small square on the diagonal of the self-similarity matrix. Other segmentation approaches include information-theoretic methods [5].

Here, we present a method to compute segmentation boundaries using a shortest path algorithm. Our assumption is that the cost of a segmentation is the sum of the individual costs of segments, and we show that with this assumption, the problem can be solved efficiently to optimality. The method is applied to popular Chinese music.

In order to optimize the manual segmentation, it is done using a particular Chinese notation system as support. This system is briefly introduced here.

2 FEATURE EXTRACTOR

The features used in the segmentation task should pinpoint the main characteristics that define the difference between two segments in a rhythmic song. While spectral characteristics generally capture the timbral evaluation well, a rhythmic feature has been chosen here. This is believed to encompass changes in instrumentation and rhythm, while not prioritizing singing and solo instruments that are liable to have influence outside the strict borders of a segment. The feature extractor consists of three steps, a note-onset detector, a rhythm feature, the rhythmogram, and a self-similarity measure of the rhythmogram.

2.1 Note onset detection

The beat in music is often marked by transient sounds, e.g. note onsets of drums or other instrumental sounds. Onset positions may correspond to the position of a beat, while some onsets fall off beat. The onset detection is made using a feature estimated from the audio, which can subsequently be used for the segmentation task. The note-onset detection aim is to give an estimate of the note onset of an instrument. This task is more difficult in the presence of other music and musical instruments with soft attacks. Often note onset methods use some sort of loudness, timbral, or similar feature. Jensen [1] compared a large number of features using an annotated database of twelve songs, and found the perceptual spectral flux (psf) to perform best. This is calculated as

$$psf(n) = \sum_{k=1}^{N_b/2} W(f_k)\left(\left(a_k^n\right)^{/3} - \left(a_k^{n-1}\right)^{/3}\right), \qquad (1)$$

where n is the feature block index, N_b is the block size, and a_k^n and f_k are the magnitude and frequency of the bin k of the Short-Time Fourier Transform (STFT), obtained using a Hanning window. The step size is 10 milliseconds, and the block size is 46 milliseconds. W is the frequency weighting used to obtain a value closer to the human loudness contour, and the power function is used to simulate the intensity-loudness power law. The power function furthermore reduces the random amplitude variations. These two steps are inspired from the PLP front-end [5] used in speech recognition.

Note onset detection systems are often used as a step in beat following. Goto and Muraoka [7] presented a beat tracking system, where two features were extracted from the audio, based on the frequency band of the snare and bass drum. Scheirer [8] took another approach, by using a non-linear operation of the estimated energy of six band-pass filters as features. The result was combined in a discrete frequency analysis to find the underlying beat. As opposed to the approaches described so far Laroche [9] built an offline system, using one feature, the energy flux, cross-correlation and dynamic programming, to estimate the time-varying tempo.

2.2 Rhythm model

The PSF feature detects most of the manual note onsets correctly, but it still has many peaks that do not correspond to note onsets, and many note onsets do not have a peak in the PSF. In order to obtain a more robust rhythm feature, the autocorrelation of the feature is now calculated on overlapping blocks of 8 seconds, with half a second step size (2 Hz feature sample rate).

$$rg_n(i) = \sum_{j=2n/fsr+8/fsr-i}^{2n/fsr+8/fsr} psf(j) \cdot psf(j+i) \qquad (2)$$

where fsr is the feature sample rate, and n is the block index. Only the information between zero and two seconds is retained. The autocorrelation is normalized so that autocorrelation at zero lag equals one.
If visualized with lag time on the y-axis, time position on the x-axis, and the autocorrelation values visualized as colours, it gives a fast overview of the rhythmic evolution of a song (figure 1).

This representation, called rhythmogram [1], provides information about the rhythm and the evolution of the rhythm in time. The autocorrelation has been chosen, instead of the FFT, for two reasons. First, it is believed to be more in accordance with the human perception of rhythm [10], and second, it is believed to be more easily understood visually.

2.3 Self-Similarity

To obtain a better representation of the similarity of song segments, a measure of self-similarity is used.

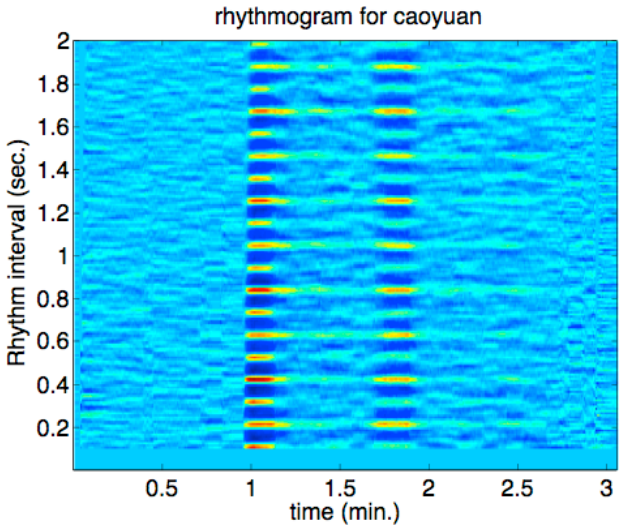

Figure 1. Rhyhtmogram of 'caoyuan'.

Several studies have used a measure of self-similarity, or recurrence plots as it was initially called [11], in automatic music analysis. Foote [1] used the dot product on MFCC sampled at a 100 Hz rate to visualize the self-similarity of different music excerpt. Later he introduced a checkerboard kernel correlation as a novelty measure [2] that identifies notes with small time lag, and structure with larger lags with good success. Bartsch and Wakefield [3] used the chroma-based representation (all FFT bins are put into one of 12 chromas) to calculate the cross-correlation and to identify repeated segments, corresponding to the chorus, for audio thumbnailing. Peeters [12] calculated the self-similarity from the FFT on the energy output of an auditory filterbank. Jensen [1] used filtering and segment following in the scale-space domain, inspired from image segmentation, to permit large-scale segmentation from small-scale self-similarity measures.

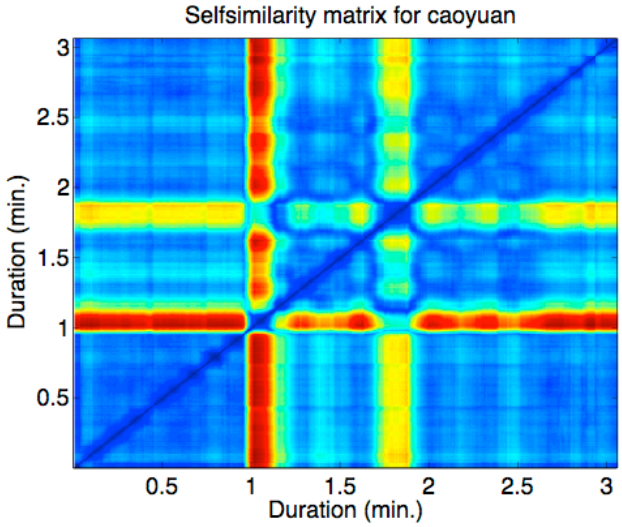

Figure 2. Selfsimilarity matrix of '草原-caoyuan'. Dark regions indicate very similar or dissimilar segments.

In this work, self similarity is calculated as the L_2 norm between the rhythmogram vectors *l* and *k*,

$$A_{lk} = \sqrt{\sum_{i=1}^{8/fsr} \left(rg_l(i) - rg_k(i)\right)^2} \quad (3)$$

In parts of the music where the rhythmic content is similar, the corresponding self similarity is close to zero. An example of the self-similarity, for the song "草原-caoyuan", can be seen in figure 2.

3 SEGMENTATION BY SHORTEST PATH

To find a best possible segmentation, based on the self-similarity matrix calculated from a model of rhythm, we first present a model for segmentation. Then we show that the problem can be solved optimally by computing a shortest path in a directed acyclic graph.

Our segmentation model is as follows. We have a sequence *1, 2, ..., N* of *N* blocks of music that should be divided into a number of segments. Let *c(i,j)* be the *cost* of a segment from block *i* to block *j*, where $1 \le i \le j \le N$. The cost of a segment should be a measure of the self-similarity of the segment, such that segments with a high degree of self-similarity have a low cost. We have chosen to use the following cost function for segments:

$$c(i,j) = \frac{1}{j-i+1} \sum_{k=i}^{j} \sum_{l=i}^{k} A_{lk} \quad (4)$$

This cost function computes the sum of the average self-similarity of each block in the segment to all other blocks in the segment. While a normalization by the square of the segment length (*j-i+1*) would give the true average, it is believed that this would severely impede the influence of new segments with larger self similarity in a large segment, since the large values would be normalized by a relatively large segment length.

Let $i_1j_1, i_2j_2, ..., i_Kj_K$ be a segmentation into *K* segments, where $i_1=1$, $i_2=j_1+1$, $i_3=j_2+1$, ..., $j_K=N$. The total cost of this segmentation is the sum of segment costs plus an additional cost *f(K)*, which depends on the number of segments:

$$E = f(K) + \sum_{n=1}^{K} c(i_n, j_n) \quad (5)$$

If we set $f(K)=\alpha K$, where $\alpha>0$ is a fixed cost for a new segment, the total segmentation cost becomes:

$$E = \sum_{n=1}^{K} \left(\alpha + c(i_n, j_n)\right) \quad (6)$$

By increasing α we decrease the number of resulting segments. Choosing an appropriate value of α is discussed in our section on experiments.

In order to compute a best possible segmentation, we construct an edge-weighted directed graph *G=(V,E)*. The set of nodes is *V={1,2,...,N+1}*. For each possible segment *ij*, where $1 \le i \le j \le N$, we have an edge *(i,j+1)* in *E*. The weight of the edge *(i,j+1)* is $\alpha+c(i,j)$. A path in *G* from node *1* to node *N+1* corresponds to a complete segmentation, where each edge identifies the individual segments. The weight of the path is equal to the total cost of the corresponding segmentation. Therefore, a shortest path (or path with minimum total weight) from node *1* to node *N+1* gives a segmentation with minimum total cost. Such a shortest path can be computed in time $O(|V|+|E|)=O(N^2)$, since *G* is acyclic and has $|E|=O(N^2)$ edges [13]. An illustration of the directed acyclic graph for a short sequence is shown in figure 3.

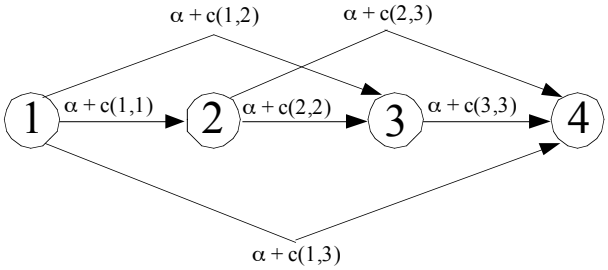

Figure 3. Directed acyclic graph for the problem of segmenting N=3 segments.

Goodwin and Laroche [14] presented a similar approach where dynamic programming was used to compute an optimal segmentation. However, in their approach the graph has N^2 nodes. As a consequence, the cost structure was less flexible, and it did not involve a parameter similar to our new segments cost (α).

4 EXPERIMENT

To test the segmentation algorithm, we have compiled a small song database including 21 popular Chinese songs coming from Chinese Mainland, Taiwan and Hong-Kong. The songs are randomly selected from Chinese top500 popular songs and have a variety in tempo, genre and style, including pop, rock, lyrical and folk. The sampling frequency of the songs is 44.1 kHz, stereo channel and 16 bit per sample.

4.1 Segmentation behaviour as a function of α

The complete segmentation system is now complete. It consists of a feature extractor (the psf), a rhythm model (the rhythmogram), a selfsimilarity measure, and finally the segmentation, based on a shortest path algorithm. The new segment cost (α) of the segmentation algorithm is analyzed here. What we are interested in is mainly to investigate if the total cost of a segmentation (eq. 6) has a local minimum. The total cost as a function of the new segment cost α is shown in figure 4. The total cost for a very small α is close to zero. As expected, the cost increases monotonically with α.

Another interesting parameter is the total number of segments. It is plausible that the segmentation system is

to be used in a situation where a given number of segments is wanted. The number of segments as a function of new segment cost α is shown in figure 5. This number decreases with the segment cost, as expected.

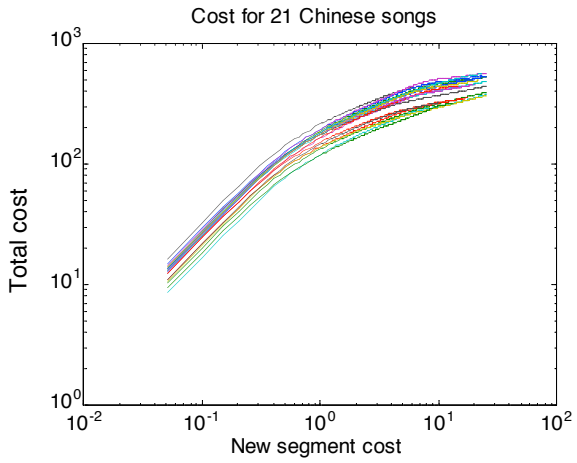

Figure 4. Total cost as a function of segment cost (α).

The number of segments follows the same shape for all songs. Indeed, the ratio between the maximum number of segments to the minimum number of segments is close to two for the main part of α and always below three. Thus, for a given α, all songs could be expected to have between, e.g. 8 and 16-24 segments.

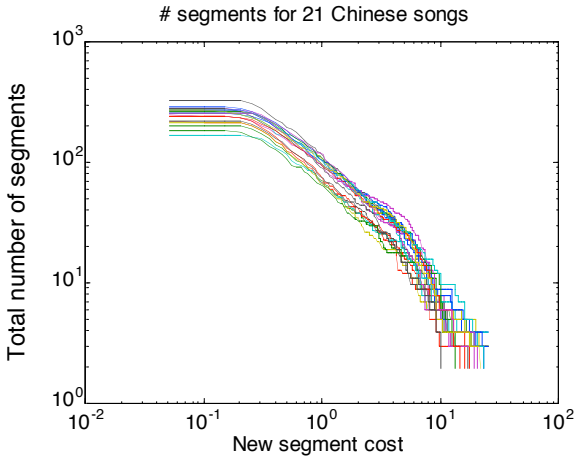

Figure 5. Total number of segments as a function of new segment cost (α).

4.2 Manual Segmentation of Popular Chinese music

In our experiment, each song is segmented manually using Audacity software to obtain a rhythm sequence such as ABCBC, where each letter identifies a similar part of the song (figure 6). A composer and an audio researcher made the manual segmentation.

Segmentation has a perceptual and subjective nature. Manual segmentation can be due to the different dimensions of music, such as rhythm or harmony. In our experiment, we limited the manual segmentation to be based on rhythm changes mainly. Most segmentation depends on different instrument and beat modulation, sometimes depends on pitch change (modulation) and rarely on the voice. We use the Chinese numbered musical notation to assist in the manual segmentation.

Figure 6 shows part of the audio for the song of "波丝猫-Bosimao" and figure 7 shows its Chinese numbered musical notation.

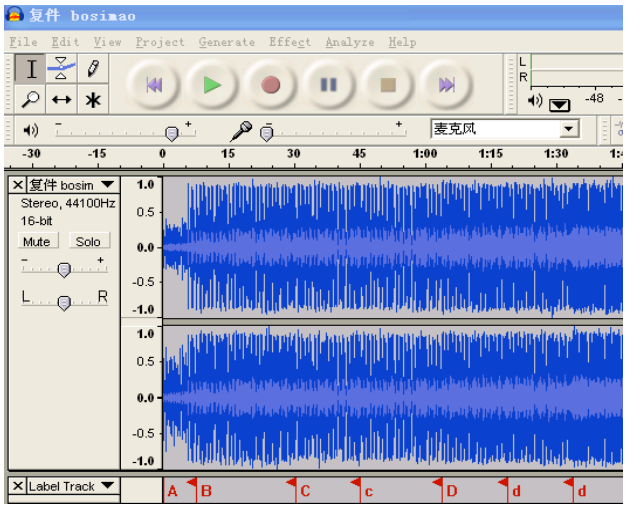

Figure 6. Manual segmentation using Audacity software.

From figure 6, we found that this part contains 4 different rhythm segment. The *A* and *B* segments are preludes played with different instruments. *c* is a repetition of the *C* segmentation, they are the same rhythm in beat but there is a clear rhythm segmenting point. The duration of *c* is longer than *C*. There is different in the end of segmentation *c*, as it includes a transition with different voice but the same beat.

In the Chinese numbered musical notation (figure 7), the tonality and time measure is marked first. Then, the melody notes are given with numbers from 1 to 7, with accompanying b/#, if applicable. The number 0 indicates silence. One dot above/below the number indicates a raise or lowering of one octave. Each number corresponds to one beat of the bar. One or several lines under the number divide the length of the note accordingly. To extend a note to more than one beat, a dash is noted at the next time location. Most other aspects of the notation, such as the bars, repetitions, and ties are notated as in the traditional Western notation. Using the notation to make the manual segmentation is believed to have helped to make a more objectively correct segmentation. The database is used to evaluate the automatic segmentation performance.

4.3 Comparison

The last step in the segmentation is to compare the manual and the automatic segment boundaries for different

values of the new segment cost (α). To do this, the automatic segmentations are calculated for increasing values of α; a low value induces many segments, while a high value gives few segments. The manual and automatic segment boundaries are now matched, if they are closer than a threshold (currently 5 seconds). For each value of α, the relative ratio of matched manual and automatic boundaries (M_m and M_a, respectively) are found, and the distance to the optimal result is minimized:

$$d(\alpha) = \sqrt{\left(1 - M_m\right)^2 + \left(1 - M_a\right)^2} \qquad (7)$$

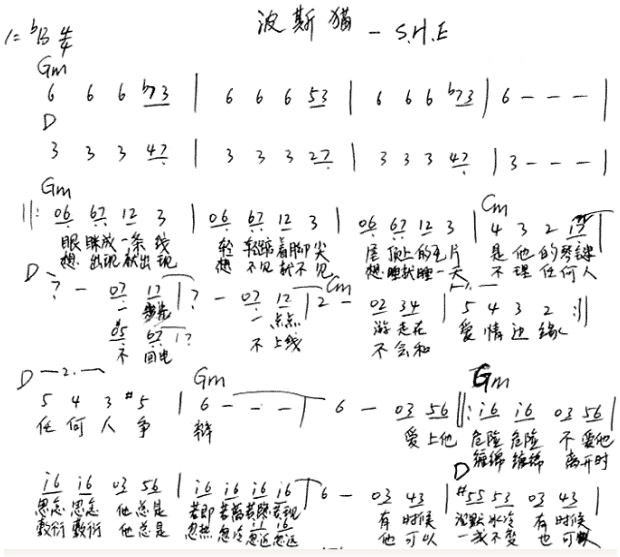

Figure 7. The Chinese numbered musical notation.

Although 5 seconds may seem a large threshold, and obviously the matched ratio diminishes with this threshold, it is chosen because it results in an appropriate number of automatic segment boundaries.

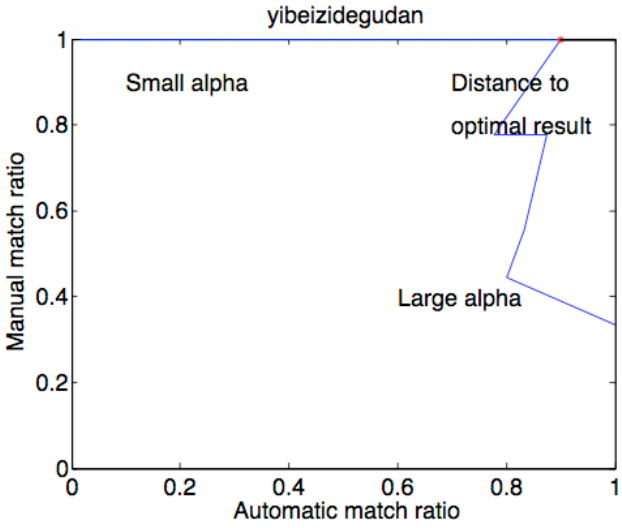

Figure 8. Manual and automatic ratio for 辈子的孤单-yibeizidegudan. A star indicates minimum distance to optimum result.

An example of this distance for ' 辈子的孤单-yibeizidegudan' is given in figure 8. The corresponding α-dependent automatic segment boundaries are shown in figure 9.The ratio between matched and total number is 8/8 (100%) for manual segmentation and 8/9 (89%) for the automatic segmentation.

The minimum distances for the optimum result for all songs are obtained for α between 4.5 and 12.9 with an average α=6.96, and one and three quartile values of 5.4 and 8.0. As a certain range of α generally gives the optimum results, approximately 10% of the α values for the different songs overlap.

4.4 Results and discussion

The comparison results of segmenting the 21 popular Chinese songs are shown in Table 1. The results show that most manual segmentation points can be detected correctly by automatic segmentation. The average ratio between matched and total numbers is 78% for manual segmentation and 74% for automatic segmentation. The average automatic ratio is 74%. These results are encouraging; in particular if the average ratios are calculated for the mean α of all songs, in which case the manual ratio is 67% and the automatic ratio is 65%.

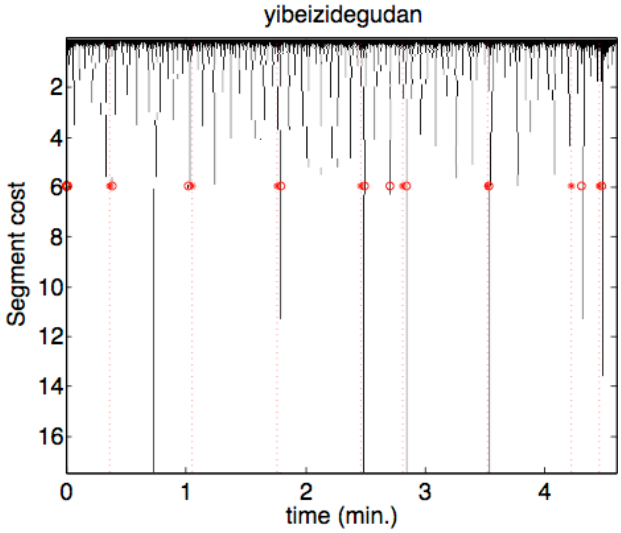

Figure 9. The automatic segment boundaries for increasing segment cost (α), with manual ('*') and automatic ('o') segment boundaries.

An example of the self-similarity, with the automatic and manual segmentation boundaries marked, is shown in figure 10. It is clear that both the automatic and manual segment boundaries are put at the beginning of a square with high similarity. Some of the automatic mark is not matched, but they do not show a important new similarity area either.

Another example is shown in figure 11. In this case, although many marks are matched, it seems that the manual marks are done on some of the segments starts,

but not on others, possible because these were just repetitions of the previous segment.

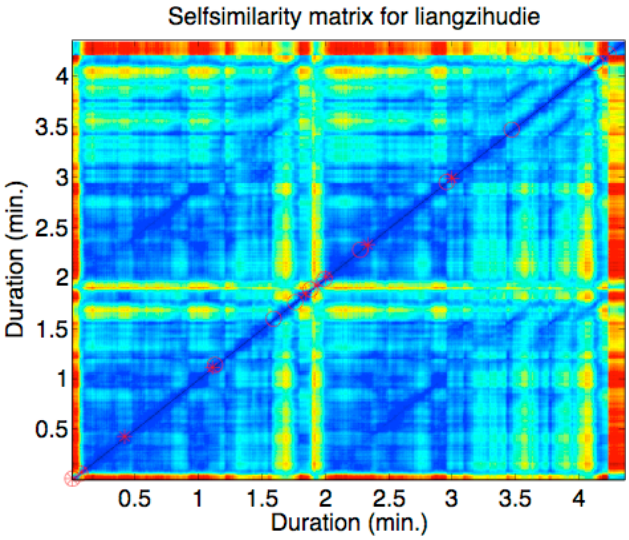

Figure 10. Self-similarity matrix for "两只蝴蝶-Liangzihudie", with manual ('*') and automatic marks ('o').

In two of the songs, “很爱很爱你-henaihenaini” and “两只蝴蝶-liangzihuidie”, the beat is not marked at all. Since the songs do not have a clear beat, the segmentation is made mainly from the voice of the songs. The manual / automatic ratio is 75%/100% and 80%/89%. This shows that the model can be used, even for music that does not have a clear beat.

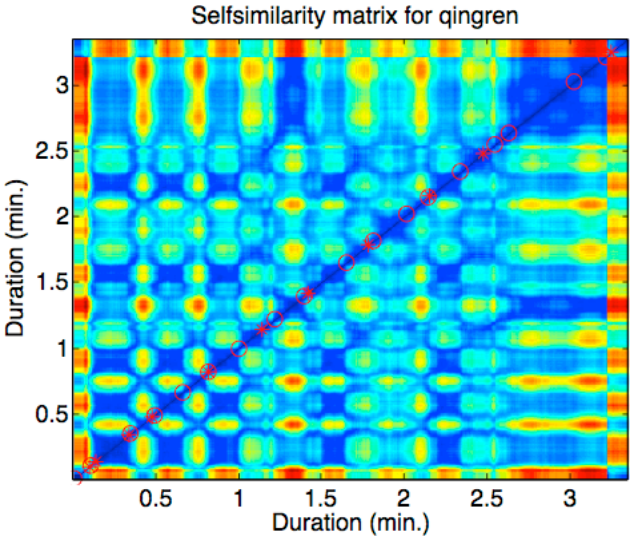

Figure 11. Self-similarity matrix for "情人-qingren", with manual ('*') and automatic marks ('o').

The automatic and manual segmentation boundaries are shown in figure 12. While it is clear that many of the segmentation boundaries are matched, some are still orphelin, i.e. not matched to another boundary. This is due both to the manual marking not being homogeneous, but some errors could also be caused by the feature and self-similarity calculation choices.

Table 1. Segmentation results of 21 popular Chinese songs

Song	Man/tot	Auto/tot
草原-Caoyuan	6/8 (75%)	6/6 (100%)
灰姑娘-Huiguniang	11/16 (69%)	11/20 (55%)
征服-Zhengfu	7/8 (88%)	7/30 (23%)
波丝猫-Bosimao	10/12 (83%)	10/17 (59%)
中国人-Zhongguoren	7/11 (64%)	7/9 (78%)
征服-Conquer	8/10 (80%)	8/12 (67%)
后来-Houlai	7/9 (78%)	7/16 (44%)
誓言-Oath2	11/14 (79%)	11/11 (100%)
情人-Qingren	9/11 (82%)	9/17 (53%)
吻别-Wenbie	16/19 (84%)	16/26 (62%)
很爱很爱你-henaihenaini	9/12 (75%	9/9 (100%)
孤单北半球-Adayan6	11/14 (79%)	11/13 (85%)
老鼠爱大米-laoshuaidami	9/10 (90%)	9/29 (31%)
爱情36计-Aq36	16/18 (89%)	16/17 (94%)
梦醒了-Mengxingle	6/9 (67%)	6/11 (55%)
辈子的孤单-yibeizidegudan	8/8 (100%)	8/10 (80%)
梦开始的地方-Mengkai	6/9 (67%)	6/6 (100%)
我们的爱-Ourlove	11/14 (79%)	11/12 (92%)
你好-Nihao	11/13 (85%)	11/12 (92%)
两只蝴蝶-Liangzihudie	8/10 (80%)	8/9 (89%)
我不是天使-Wbsts	5/9 (56%)	5/7 (71%)
average	9/12 (78%)	9/14 (74%)

5 CONCLUSIONS

This paper presents a feasible method to segment music. This is done using a rhythm model, the rhythmogram and a shortest path segmentation algorithm based on the self similarity of the rhythmogram. The free parameter of the shortest path algorithm, the cost of a new segment, has been found by minimizing the distance to a optimal solution in matching manual and automatic segmentation boundaries. Experiments using popular Chinese music have resulted in 67% manual segmentation ratio and 65% automatic segmentation ratio based on a database with 21 songs. This result is improved to 78%/75% with individually optimized segment costs for each song. There are 5 songs which automatic segmentation achieved 100% correct detection. This indicated that the rhythm-based segmentation is useful for Chinese music, but probably also for much other popular music, because of the similarities to the Chinese music.

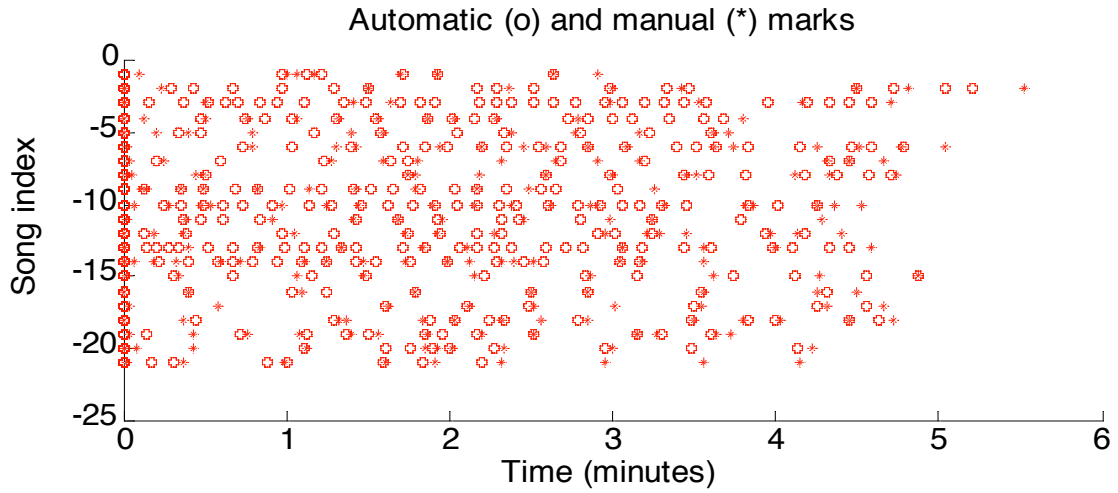

Figure 12. Automatic (o) and manual (*) segment boundaries for the 21 Chinese songs.

ACKNOWLEDGEMENT

Laurent "Saxi" Georges has been most helpful in understanding the Chinese numbering system, helping with the manual segmentation and in general music related discussions.

REFERENCES

[1] Foote, J., Visualizing Music and Audio using Self-Similarity. In Proceedings of ACM Multimedia '99, pp. 77-80, Orlando, Florida, 1999.

[2] Foote, J., Automatic Audio Segmentation using a Measure of Audio Novelty. In Proceedings of IEEE International Conference on Multimedia and Expo, vol. I, pp. 452-455, July 30, 2000.

[3] Bartsch, M. A. and Wakefield, G.H., To Catch a Chorus: Using Chroma-Based Representations for Audio Thumbnailing. *in* Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 15-18, 2001.

[4] Jensen K., A Causal Rhythm Grouping. Lecture Notes in Computer Science, Volume 3310, pp. 83-95. 2005.

[5] Dubnov, S., Assayag, G., El-Yaniv, R., Universal Classification Applied to Musical Sequences. Proc. of the International Computer Music Conference, Ann Arbour, Michigan, pp. 332-340 1998.

[6] Hermansky H., Perceptual linear predictive (PLP) analysis of speech, J. Acoust. Soc. Am., vol. 87, no. 4, pp. 1738-1752, Apr. 1990.

[7] Goto M., and Muraoka, Y., A real-time beat tracking system for audio signals. Proceedings of the International Computer Music Conference, pp. 171-174, 1995.

[8] Scheirer, E., Tempo and Beat Analysis of Acoustic Musical Signals, Journal of the Acoustical Society of America, Vol. 103, No. 1, pp. 588-601, 1998.

[9] Laroche J., Efficient tempo and beat tracking in audio recordings, J. Audio Eng. Soc., 51(4), pp. 226-233, April 2003.

[10] Desain P., A (de)Composable theory of rhythm. Music Perception, 9(4) pp. 439-454, 1992.

[11] Eckmann, J. P., Kamphorst, S. O., and Ruelle, D., Recurrence plots of dynamical systems, Europhys. Lett. 4, 973, pp. 973-977, 1987.

[12] Peeters, G., Deriving musical structures from signal analysis for music audio summary generation: sequence and state approach. *In* Computer Music Modeling and Retrieval (U. K. Wiil, editor). Lecture Notes in Computer Science, LNCS 2771, pp. 143-166, 2003.

[13] Cormen T. H., Stein C., Rivest R. L., Leiserson C. E., *Introduction to Algorithms*, Second Edition. The MIT Press and McGraw-Hill Book Company, 2001.

[14] Goodwin, M. M. and Laroche, J., Audio segmentation by feature-space clustering using linear discriminant analysis and dynamic programming. In Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 131-134, 2003.